CMPT 215.3 IN-LAB EXAMINATION
(Group B)
October 30, 2000

Total Marks: 10   OPEN BOOK and OPEN NOTES
Time: 90 minutes (5:00pm-6:30pm)

Instructions

Both questions are programming questions. You'll need to design algorithms, write, debug, and test MIPS assembly programs. The marks for each question are as indicated. Allocate your time accordingly.

1. **(5 marks)** Read in from the keyboard a sequence of **non-zero** integers finished by a 0, and print the number of positive integers and the number of negative integers, respectively. Note that the last 0 does not belong to this sequence of integers. For example, if the user inputs the following six numbers: 10, 3, -17, -4, 9, 0, the output should be:

> *There are 3 positive integers*
> *There are 2 negative integers*

2. **(5 marks)** Implement function **int *replace*(int *data*, int *new_data*, int *array_addr*, int *array_len*)** that searches the first occurrence of the *data* in an integer array, replaces it with the *new_data*, and returns the index of the *data* in the array. If the *data* is not found, -1 is returned instead. Note that the index of the array starts at 0. In the function, the *array_addr* is the starting address of the integer array, and the *array_len* is the total number of integers in the array. For example, assume the original array is 7, 24, 15, 3, 15, 10, 9, if you replace 15 by 34, the new array will be 7, 24, 34, 3, 15, 10, 9.

In your *main*, you should test the function *replace* by first creating an array with seven integers 7, 24, 15, 3, 15, 10, 9, and then calling the function as follows:

> *index1 = replace(15, 34, array_addr, array_len)*
> *index2 = replace (7, 11, array_addr, array_len)*
> *index3 = replace (9, 2, array_addr, array_len)*

After each replacing, print the index returned by the function, and the resulted new array. Note that " " (space) should be used to separate array elements from each other.